

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

RECEIVED
CENTRAL FAX CENTER

JUN 29 2004

OFFICIAL

AMENDMENTS TO THE CLAIMS

Claims 1-40 were originally pending.

Please cancel claims 1, 10, and 18-32 without prejudice.

Please amend claims 2-8, 11-16, and 33-39.

No claims are added.

Accordingly, claims 2-9, 11-17, and 33-40 remain pending.

The following listing of claims replaces all prior versions, and listings of claims in the application.

Listing of Claims:

1. (Canceled).
2. (Currently amended) A method as recited in claim 74, wherein the directory schema is independent from any values that the object instance may store.
3. (Currently amended) A method as recited in claim 74, wherein the complex data type is an XML data type.
4. (Currently amended) A method as recited in claim 74, wherein the first property is described by an XML string, the XML string being compatible with the complex data type.

1
2 5. (Currently amended) A method as recited in claim 74, wherein the
3 first property is described by an XML string, the XML string being compatible
4 with the complex data type.

5
6 6. (Currently amended) A method as recited in claim 74, wherein the
7 directory schema defines a plurality of object classes for a versioning aware
8 directory.

9
10 7. (Currently amended) A method as recited in claim 1, wherein the
11 ~~object instance is a first object instance, wherein the property is the first property~~
12 ~~comprising a particular number of data elements, and wherein the method further~~
13 ~~comprises:~~ for extending a directory schema independent of schema modification,
14 the method comprising:

15 instantiating a first object instance of a content class comprising a flexible
16 attribute that is based on a complex data type, the first object instance being
17 created in a directory based on the directory schema, the content class and flexible
18 attribute being defined in the directory schema;

19 assigning a first property to the attribute, the first property being
20 operational or data providing in nature, the property having a data type that is
21 independent of the complex data type; and

22 independent of directory schema modification;

23 (a) instantiating a second object instance of the content class, the second
24 object comprising a second flexible attribute that is based on the complex data
25 type;

1 (b) assigning a second property to the second flexible attribute, the second
2 property having a different number of data elements than the first property; and

3 (c) wherein each data element in the first and second properties comprises
4 data such neither the first or second properties contribute to data sparsity of a
5 directory based on the directory schema.

6
7 8. (Currently amended) A method ~~as recited in claim 1, wherein the~~
8 ~~object instance is a first object instance, wherein the property is the first property,~~
9 ~~and wherein the method further comprises:~~ for extending a directory schema
10 independent of schema modification, the method comprising:

11 instantiating a first object instance of a content class comprising a flexible
12 attribute that is based on a complex data type, the first object instance being
13 created in a directory based on the directory schema, the content class and flexible
14 attribute being defined in the directory schema;

15 assigning a first property to the attribute, the first property being
16 operational or data providing in nature, the property having a data type that is
17 independent of the complex data type;

18 without modifying the directory schema, instantiating a second object
19 instance of the content class, the second object comprising a second flexible
20 attribute that is based on the complex data type; and

21 assigning a second property to the second flexible attribute, the second
22 property being completely independent of any operational or data characteristics
23 of the first property.

1 9. (Original) A method as recited in claim 8, wherein the first and
2 second properties are described by respective XML strings that are compatible
3 with the complex data type.

4
5 10. (Canceled).

6
7 11. (Currently amended) A server as recited in claim 15 ~~10~~, wherein the
8 directory schema is independent from any values that the object instance may
9 store.

10
11 12. (Currently amended) A server as recited in claim 15 ~~10~~, wherein the
12 complex data type is an XML data type.

13
14 13. (Currently amended) A server as recited in claim 15 ~~10~~, wherein the
15 first property is described by an XML string, the XML string being compatible
16 with the data type.

17
18 14. (Currently amended) A server as recited in claim 15 ~~10~~, wherein the
19 directory schema defines a plurality of object classes for a versioning aware
20 directory.

15. (Currently amended) A server comprising: as recited in claim 10;
wherein the object instance is a first object instance, wherein the property is the
first property comprising a particular number of data elements, and wherein the
computer-executable instructions further comprise instructions for:

a processor coupled to a memory, the memory comprising computer
executable instructions, the processor being configured to fetch and execute the
computer-executable instructions for:

instantiating a first object instance of a content class comprising a flexible
attribute that is based on a complex data type, the content class and flexible
attribute being defined in a directory schema; and

assigning a first property to the attribute, the first property comprising a
particular number of data elements, the first property being operational or data
providing in nature, the property having a data type that is independent of the
attribute's data type; and

independent of directory schema modification:

(a) instantiating a second object instance of the content class, the
second object comprising a second flexible attribute that is based on the complex
data type;

(b) assigning a second property to the second flexible attribute, the
second property having a different number of data elements than the first property;
and

(c) wherein each data element in the first and second properties
comprises data such neither the first or second properties contribute to data
sparsity of a directory based on the directory schema.

1 16. (Currently amended) A server comprising: as recited in claim 10,
2 wherein the object instance is a first object instance, wherein the property is the
3 first property, and wherein the computer-executable instructions further comprise
4 instructions for:

5 a processor coupled to a memory, the memory comprising computer
6 executable instructions, the processor being configured to fetch and execute the
7 computer-executable instructions for:

8 instantiating a first object instance of a content class comprising a flexible
9 attribute that is based on a complex data type, the content class and flexible
10 attribute being defined in a directory schema; and

11 assigning a first property to the attribute, the first property comprising a
12 particular number of data elements, the first property being operational or data
13 providing in nature, the property having a data type that is independent of the
14 attribute's data type;

15 without modifying the directory schema, instantiating a second object
16 instance of the content class, the second object comprising a second flexible
17 attribute that is based on the complex data type; and

18 assigning a second property to the second flexible attribute, the second
19 property being completely independent of any operational or data characteristics
20 of the first property.

21
22 17. (Original) A server as recited in claim 16, wherein the first and
23 second properties are described by respective XML strings that are compatible
24 with the complex data type.
25

1 18-32. (Canceled).

2
3 33. (Currently amended) A server as recited in claim 38 32, wherein the
4 directory schema is independent from any values that the object instance may
5 store.

6
7 34. (Currently amended) A server as recited in claim 38 32, wherein the
8 complex data type is an XML data type.

9
10 35. (Currently amended) A server as recited in claim 38 32, wherein the
11 first property is described by an XML string, the XML string being compatible
12 with the complex data type.

13
14 36. (Currently amended) A server as recited in claim 38 32, wherein the
15 first property is described by an XML string, the XML string being compatible
16 with the complex data type.

17
18 37. (Currently amended) A server as recited in claim 38 32, wherein the
19 directory schema defines a plurality of object classes for a versioning aware
20 directory.

1 38. (Currently amended) A server ~~as recited in claim 32, wherein the~~
2 ~~object instance is a first object instance, wherein the property is the first property~~
3 ~~comprising a particular number of data elements, and wherein the server further~~
4 ~~comprises processing means for:~~ for extending a directory schema independent of
5 schema modification, the server comprising:

6 processing means for:

7 instantiating a first object instance of a content class comprising a
8 flexible attribute that is based on a complex data type, the content class and
9 flexible attribute being defined in a directory schema; and

10 assigning a first property to the attribute, the first property being
11 operational or data providing in nature, the first property having a data type that is
12 independent of the complex data type, the first property comprising a particular
13 number of data elements; and

14 independent of directory schema modification:

15 (a) instantiating a second object instance of the content class,
16 the second object comprising a second flexible attribute that is based on the
17 complex data type;

18 (b) assigning a second property to the second flexible
19 attribute, the second property having a different number of data elements than the
20 first property; and

21 (c) wherein each data element in the first and second
22 properties comprises data such neither the first or second properties contribute to
23 data sparsity of a directory based on the directory schema.

1 39. (Currently amended) A server ~~as recited in claim 32, wherein the~~
2 ~~object instance is a first object instance, wherein the property is the first property,~~
3 ~~and wherein the server further comprises processing means for:~~ for extending a
4 directory schema independent of schema modification, the server comprising:

5 processing means for:

6 instantiating a first object instance of a content class comprising a
7 flexible attribute that is based on a complex data type, the content class and
8 flexible attribute being defined in a directory schema; and

9 assigning a first property to the attribute, the first property being
10 operational or data providing in nature, the first property having a data type that is
11 independent of the complex data type, the first property comprising a particular
12 number of data elements;

13 independent of modification to the directory schema, instantiating a second
14 object instance of the content class, the second object comprising a second flexible
15 attribute that is based on the complex data type; and

16 assigning a second property to the second flexible attribute, the second
17 property being completely independent of any operational or data characteristics
18 of the first property.

19
20 40. (Original) A server as recited in claim 39, wherein the first and
21 second properties are described by respective XML strings that are compatible
22 with the complex data type.
23
24
25